A.J.H. Morrell (Ed.), Information Processing 68.2. Proceedings of IFIP Congress, 1968, Edinburgh. Amsterdam: North-Holland, pp. 1103-1113.

INFORMATION PROCESSING 68 - NORTH-HOLLAND PUBLISHING COMPANY - AMSTERDAM (1969)

COMPUTER-ASSISTED INSTRUCTION: AN OVERVIEW OF OPERATIONS AND PROBLEMS*

Patrick SUPPES
Institute for Mathematical Studies in the Social Sciences
Stanford University, Stanford, California, USA

This paper is mainly concerned to give an overview of the programs currently running at Stanford in computer-assisted instruction. These are the drill-and-practice program in elementary mathematics, the tutorial mathematics program for grade 2, the tutorial reading program for grade 1. the tutorial logic and algebra program for grades 5 to 8, the tutorial Russian program at the university level and an audio program using computer-generated speech and home telephones as terminals. Some of the most important problems that must be faced in the future development of computer-assisted instruction are discussed.

The spread of technology and the mechanisms of technological change are, as yet, little understood. From the serious use of block printing in the tenth century in China, to the widespread use of books in schools in the Western world during the latter part of the eighteenth century, more than 800 years elapsed. Moreover, the work of editing and of printing the Confucian classics in the tenth century lasted for 21 years, a period longer than computers have as yet been available. But the analogy that gives pause is not the 21 years but the 800 years before books were used on a widespread basis. On my more pessimistic days, after looking at the total cost of computer systems suitable for providing instruction in schools, I wonder if it will not be at least well into the twenty-first century before the economics of education will sanction the widespread use of computers in schools. However, in spite of the fact that we are not yet on the verge of universal use of computers in education for instruction, we now have several years of operating experience, and there are concrete plans underway in the United States for introducing computer terminals as instructional devices in a number of school districts, colleges and universities.

The technological development of computer-assisted instruction falls naturally into three stages. In the beginning, concentrated efforts are made at major sophisticated centers of research that have available a great depth and variety of computer talent. At the second stage, demonstration efforts are to be found in school systems and colleges that may not, themselves, have major computer resources but are concerned to lead the way in innovation and educational change. These demonstration efforts are not intended to reach the bulk of the students taught by the school system or college, but to serve as first steps in gaining experience and understanding of the possibilities and limitations, both technologically and economically, of computer-assisted instruction. At the third stage, which has not yet been reached anywhere in the world, widespread use of computers for instructional purposes takes place, and entire school systems or colleges saturate given areas of the curriculum with computer terminals serving as instructional devices. I would say that in the United States we are now somewhere between the first and the second stage. A number of efforts in computer-assisted instruction in various university centers in the United States are known to most of you, and I shall not attempt to list them here. During the next 18 months, probably some 20 to 40 school systems will also begin actual operation of computer systems for instructional purposes.

To give you a sense of the kind of operations that are now being explored, I would like to de-

* The research reported here has been supported by the Carnegie Corporation of New York, the U.S. National Science Foundation and the U.S. Office of Education. My indebtness to my many younger collaborators is very great indeed. Professors Atkinson and Van Campen are mentioned in the text. In addition I must mention as a minimum: Fred Binford, Dow Brian, Marilyn Cox, Jamesine Friend, John Gwinn, Richard Hull, Max Jerman, Marvin Levin and Roulette Smith.

scribe for you the Stanford programs in computer-assisted instruction in operation during the academic year 1967-68. I would emphasize again that a number of other universities are also engaged in such efforts, and by concentrating on the details of the efforts at Stanford, I do not mean to make any special claims for the uniqueness of our efforts. It is just that by attempting a reasonably detailed overview of the Stanford operations, I believe I can give you a concrete sense of what computer-assisted instruction means in practice and what some of its problems are.

I have summarized these Stanford programs in table 1. The number of students shown for each type of program indicates the number of students engaged on a daily basis throughout the academic year, and the type of terminal used is also shown. (TTY stands for teletype and CRT for cathode-ray tube with keyboard and lite-pen.) As the figures show, we have been processing more than 2 000 students daily, but at least as important as that number is the variety of programs we have been operating. The drill-and-practice mathematics program, the tutorial logic and algebra program, the tutorial Russian program, and the dial-a-drill program, which together encompass the bulk of the students, are all run simultaneously on the PDP-1 computer complex we have built up since 1963, initially with the close cooperation of Professor John McCarthy and his group of young computer scientists. The tutorial mathematics and reading programs are run on a separate system, an IBM 1500 system, located on the school grounds a few miles from Stanford. The school programs run on the PDP-1 are run from the computer on the campus and reach the schools via telephone lines. Some of these lines reach a great distance. We have on-line operations more than 1500 miles away in Kentucky and in Mississippi. In both cases there is on site a PDP-8 that serves primarily as a line concentrater and multiplexer, and we have between Stanford and the PDP-8, in each of these states, a single telephone line, capable of handling up to 60 teletype terminals. At the present time we have 20 teletype terminals in Mississippi and 32 in Kentucky. This number is scheduled to go to 60 terminals in each of the two places in the fall of this year.

Perhaps the single most powerful argument for computer-assisted instruction is the potential it offers in terms of individualizing instruction. The basic idea of individualization is simple and straightforward. It is to attend to the needs of each student on a real-time basis, to offer curriculum material, and to respond to this work in a style that can be ideally represented by that of a sensitive and gifted tutor. No one thinks for a moment that in the immediate future we shall be able to do all of this on a computer. I would like to give you a concrete sense of the promise and the limitations of our current efforts.

As I have already indicated, the individualized drill-and-practice program in elementary mathematics is our largest program in terms of number of students processed, number of terminals, and geographical distribution of terminals. Some sample exercises for children of various ages in the elementary school are shown in fig. 1. These exercises are arranged so that those at the top are the sort given to students in the first grade (six- or seven-year olds) and extend to those at the bottom which are given to students in the sixth grade (twelve-year olds). You can see from these examples how the students are expected to respond. They use the keyboard of the teletypes but ordinarily they need use only one or two keys to make a numerical response or to input a special symbol like >. Above all, we do not expect elementary-school children to learn to type in ordinary secretarial fashion. I can confidently say on

Table 1
Stanford 1967-68 programs in computer-assisted instruction

Program	Students	Terminals
Drill-and-practice mathematics, Grades 1-8		
California	985	TTY
Kentucky	810	TTY
Mississippi	592	TTY
Tutorial mathematics, Grade 2	76	CRT + Audio + Film
Tutorial reading, Grade 1	78	CRT + Audio + Film
Tutorial logic and algebra, Grades 5-8	195	TTY
Tutorial Russian, University level	30	TTY + Audio
Dial-a-drill	15	Telephone

```
COUNT

0 _1_ 2 3

1 2 _3_ 4

2 _3_ 4 5

5 _6_ 7 8

4 5 6 _7_
                                        GRADE 1
```

```
BETTY HAS 1 BICYCLE, 2 ROLLER SKATES, 8 GAMES,
AND 3 DOLLS. HOW MANY TOYS DOES BETTY HAVE...

 _14 TOYS
                                        GRADE 2
TYPE < OR = OR >
8 + 8 _>_ 5 + 7

IT IS 11:30 NOW. WHAT TIME WAS IT ONE HOUR AND THIRTY
MINUTES AGO...

 _10 O 'CLOCK
```

```
69 / 6 0 1 9
                    _1_
NO, TRY AGAIN
                                        GRADE 5
69 / 6 0 1 9
                    _2_
NO, ANSWER IS 80, TRY AGAIN
```

```
2/3 X 3/7 = _6_/21

          = _2_/ _7_

4/9 DIVIDED BY 7/3 = 4/9 X _3_/ _7_

                   = _12/63
                                        GRADE 6
                   = 4/ _21

5 1/2 + 17 2/3 = ( _5_ + 17) + ( _3_/6 + _4_/6)

               = _22 + _7_/6

               = _23 _1_/6
```

```
THIS IS A REVIEW ON THE LAWS OF ARITHMETIC.

TYPE A IF THE OPERATION IS ASSOCIATIVE.
TYPE C IF THE OPERATION IS COMMUTATIVE.
TYPE D IF THE OPERATION IS DISTRIBUTIVE.
TYPE S IF SOME OTHER RULE IS USED.

54 / (9 / 3) = (54 / 9) X 3
 _C_
NO, TRY AGAIN
                                        GRADE 6
 _D_
NO, ANSWER IS S. TRY AGAIN

 _S_

64 / (8 / 2) = (64 / 8) X 2
 _S_

66 X 56 = 56 X 66
 _C_
```

Fig. 1. Examples of exercises from the individualized drill-and-practice program in elementary mathematics.

the basis of more than three years' experience that it is not difficult for even the youngest children in elementary school to learn to use a teletype in the mode required by these exercises. The program exemplified by the exercises shown in fig. 1 is individualized in at least three ways. First, and perhaps most importantly, the student is immediately told whether his answer is correct. If he does not answer in time he is told that time is up, and the problem is repeated. This immediate correction procedure brings with it a briskness of pace that is often missing from classroom teaching. Secondly, the exercises are given to students at a level of difficulty commensurate with their level of achievement, as measured by their past performance on the preceding day. In the main program that has been run during the academic year 1967-68, students began a concept block by taking a pre-test. On the basis of the pre-test they are put in one of five levels of difficulty and then are moved upward or downward or stay at the same level depending upon their daily scores. At the end of five days of training at differing levels of difficulty, all students are given the same post-test. This post-test score is kept as part of a vector of post-test scores on concept blocks, and is used for selection of the third aspect of individualization. When a student is working on a current concept, work on that concept constitutes about 70 per cent of his program. The other 30 per cent is devoted to individualized review on that past concept block on which his performance was the worst. We scan his previous post-test scores, select the lowest, with the restriction that we not repeat the same block for review twice in a row, and give him review problems at the level of difficulty commensurate with his post-test score. At the end of the review exercises, we give him a review test and use the score on the new review test as his current score on that past concept block.

In the program that we began to implement during 1967-68 and that will be a major part of our endeavors during 1968-69, we are increasing the individualization by removing age levels and grade levels. Each major concept is organized into a strand running across six years of elementary-school mathematics. Such strand analysis of elementary-school mathematics is common in standard curriculum analysis. All the familiar and expected topics are included in one of the strands. We are arranging the material on a given strand in terms of scale of difficulty. The student begins at the bottom of the strand and works his way up according to a performance criterion. At any given time, we can give a good summary of the student's position in the elementary-school mathematics curriculum by providing a printout of his positions in the 12 to 15 strands of the curriculum. It is not possible here to review many features of this strand program. Even from the brief description I have given, however, it should be clear that the strand program does permit a high degree of individualization. Each student is now free to progress through the curriculum as rapidly as he is able. More importantly, each student is given his daily work at a level commensurate with his current level of achievement and accomplishment.

I have not provided illustrations of the curriculum material in the tutorial reading and tutorial mathematics in our Brentwood program. I should mention that the tutorial reading program is under the direction of Professor Richard C. Atkinson. We have been able to use to good advantage the display of colored films and pictures as well as the displays possible on the cathode-ray tubes. The Brentwood system is located in a culturally deprived area, and perhaps one of the most striking results we have found thus far concerns the level of achievement in the first-grade tutorial mathematics program during the academic year 1966-1967. We used as control subjects the students who were running on the system in reading but were receiving mathematics instruction in the ordinary classroom. We found that the students receiving the tutorial mathematics program did better than the control group as measured by standard off-line achievement tests given at the end of the year. However, due to the great variety of achievement and ability in both the experimental and control groups, the comparison between the two was not statistically significant. Fortunately, there were two classes in the experimental group and also in the control group and they were very evenly matched in terms of initial ability, as measured by Stanford-Binet intelligence tests. When we looked at the achievement-test data in more detail in terms of classes we found the following striking finding. The two low-ability classes, one experimental and one control, which had mean IQ's of 82.8 and 81.9, respectively, reached a very different level of achievement. The experimental group was much superior in performance, not only on the basis of a rough appraisal of the data but at a statistically significant level. Providing an introduction to mathematics at the beginning of their school experience for slower children is a very demanding and difficult teaching task. We are not really able to offer an explanation of the superior results achieved by the experimental group, but a natural conjecture is that the patient and individualized character of the computer program is responsible for producing the difference.

In the case of the many classes involved in the individualized drill-and-practice program in which a student is engaged ordinarily from 4 to 8 minutes per day and completes an average of 20 exercises, the external measures of achievement have produced more heterogeneous results. Other variables such as the amount of school and teacher effort devoted to the mathematics curriculum have affected whether the experimental classes did significantly better on achievement tests. The overall evidence is favorable; as might be expected, however, when much of the teaching is left to the teacher and when the individualized drill and practice are restricted to a relatively short period per day, it is easy for other variables to turn out to be more significant. In the case of the tutorial mathematics at grade 1, and during the past academic year at grade 2, the situation is different. In this case, we have considerably greater control over the curriculum and we would naturally expect to obtain more significant differences in comparing experimental and control groups.

I should also emphasize that we are by no means content with the very fragmentary kind of summary data analysis provided by off-line achievement tests. In fact, the main thrust of our efforts at using the data generated by the students' performance at the computer terminals has been to reach for a deeper and better understanding of how students learn elementary mathematics and to explain in clear fashion why they have the learning difficulties they do. A major thrust of our work in drill and practice has been devoted to the identification of structural variables that account fairly well for the relative difficulty of exercises. To give one simple example, in looking at two-, three-, or four-digit problems in column addition involving the sum of two numbers, we have found by applying linear-regression techniques of analysis that about 75 per cent of the variation in difficulty in the several million problems that fall within the class defined may be accounted for by two simple variables, the number of digits and the number of "carry's". It is not possible here to go more deeply into the kinds of analysis we are doing, but much information of this sort is included in Suppes, Jerman, and Brian (1968). A second book is now under preparation concerning a still more extensive analysis of data from the 1966-1967 mathematics program. Parenthetically, it should be noted that the opportunities for collecting behavioral data on students' performance in subject matter presented by computer terminals are unparalleled in the history of education and psychology. In the long run, probably one of the most important gains of computer-assisted instruction will be precisely this opportunity to collect and analyze data on an unparalleled scale. If our own experience over the past several years is any indication, the very presence of the data will itself be a major challenge to deepen our conceptions of learning and raise our aspirations to develop more powerful scientific theories of learning.

I turn next to the program in tutorial logic and algebra. This is the only program we run that is aimed mainly at very bright students. It is offered as a supplement or enrichment to the regular mathematics program of the students. We are particularly anxious to make the program as self-contained and tutorial as possible in order to bring it to elementary schools and junior high schools for students from 10 to 13 years of age who would not ordinarily be offered such material because the teaching resources in their schools would be too restricted. The objective of the course is to begin with the elements of sentential logic, move rapidly to the logic of identity and simple inferences in elementary algebra, and go on to a full set of axioms for ordered fields and to the proofs of the standard elementary theorems about ordered fields. This material covers somewhat more than two years of the curriculum, with students spending about 6 minutes a day at terminals. In the third year, we introduce the square root operation, and prove theorems about linear equations and quadratics. Fig. 2 shows some of the derivations that occur fairly early in the course. From a computer standpoint the important aspect of the logic and algebra program is that the student is always issuing commands to the computer as to what step to take next. As many people have remarked, we are in fact teaching the students a simple programming language. Thus, for example, in the proof that $7 = 4 + 3$, the student has available the associative law of addition, the commutative law of addition, and the definition of each positive integer as its predecessor plus 1. The student begins the proof in line 1 by calling for the number definition of 7 (ND7). In the second line he asks that the first occurrence of 6 be replaced by its definition in line 1 to obtain line 2 (1D6.1). In line (3) he uses the associative law to ask that association to the right be performed around the second addition symbol in line (2) (2AR2). In line (4) he calls for the replacement of $1 + 1$ by 2 (3ID2.1). Then in line (5) he calls for the replacement of 5 by its

```
DERIVE:  7=4+3

ND7          (1)  7=6+1
1D6.1        (2)  7=(5+1)+1
2AR2         (3)  7=5+(1+1)
3ID2.1       (4)  7=5+2
4D5.1        (5)  7=(4+1)+2
5AR2         (6)  7=4+(1+2)
6CA2         (7)  7=4+(2+1)
7ID3.1       (8)  7=4+3

CORRECT.
```

```
DERIVE:  9=5+4

ND9          (1)  9=8+1
1D8.1        (2)  9=(7+1)+1
27D7.1       (3)  9=((6+1)+1)+1
3D6.1        (4)  9=(((5+1)+1)+1)+1
4AR3         (5)  9=((5+1)+(1+1))+1
5AR4         (6)  9=(5+1)+((1+1)+1)
6AR3                   YOU CANNOT ASSOCIATE RIGHT
                        AROUND PLUS SIGN NUMBER 3.
6ID2.1       (7)  9=(5+1)+(2+1)
7ID3.1       (8)  9=(5+1)+3
8AR2         (9)  9=5+(1+3)
9CA2         (10) 9=5+(3+1)
10ID4.1      (11) 9=5+4

CORRECT.
```

```
DERIVE:  (3+B)+(C+D)=3+((B+C)+D)

AS                    (A+B)+C=A+(B+C)
A:3
B:B
C:C+D            (1)   (3+B)+(C+D)=3+(B+(C+D))
AS                    (A+B)+C=A+(B+C)
A:B
B:C
C:D             (2)   (B+C)+D=B+(C+D)
CE1            (3)   B+(C+D)=(B+C)+D
1.3REI         (4)   (3+B)+(C+D)=3+((B+C)+D)

CORRECT.
```

```
USE TWO OPERATIONS HERE.

MARY HAD 128 POSTCARDS.  HER MOTHER GAVE HER 17 MORE.
HER BROTHER TOOK AWAY 10 POSTCARDS.
HOW MANY POSTCARDS DID SHE HAVE LEFT...
G              (1)  128
G              (2)  17
G              (3)  10
1.2A           (4)  145
4.3S           (5)  135
5X
CORRECT.
```

```
HERE IS THEOREM 7 FOR YOU TO DERIVE

DERIVE:  A+B=A+C → B=C

WP              (1)      A+B=A+C
1CA1            (2)      B+A=A+C
2CA2            (3)      B+A=C+A
3AE
:-A             (4)      (B+A)+(-A)=(C+A)+(-A)
4AR2            (5)      B+(A+(-A))=(C+A)+(-A)
5AR4            (6)      B+(A+(-A))=C+(A+(-A))
TH2                   (-A)+A=0
A:A             (7)      (-A)+A=0
7CA1            (8)      A+(-A)=0
6.8REI          (9)      B+0=C+(A+(-A))
9.8REI          (10)     B+0=C+0
Z                     A+0=A
A:B             (11)     B+0=B
Z                     A+0=A
A:C             (12)     C+0=C
10.11REI        (13)     B=C+0
13.12REI        (14)     B=C
1.14CP          (15)     A+B=A+C → B=C

CORRECT.
```

Fig. 2. Examples of derivations taken from the tutorial program in logic and algebra.

definition (4D5.1). In line (6) another association to the right is performed (5AR2). In line (7) the commutative law is applied around the second addition sign (6CA2). Finally, in line (8), 2 + 1 is replaced by the number it defines, 3, to obtain the desired result (7ID3.1). The second derivation shown in fig. 2, is longer; it shows what happens when the student issues an illicit command to the computer. He immediately gets back an error message indicating why the rule cannot be applied in the manner that has been called for. In this second derivation it is particularly evident that it would be time-consuming and troublesome for the student himself to write out each line of proof. The commands to the computer are restricted mainly to three or four letters or numerals of input whereas the equation of line (6) has 17 characters. Moreover, it is an expression that would be troublesome even for most of us to input without mistake.

In the third derivation shown in fig. 2, an important next step is indicated. The input characters at the left indicate how the student is to apply the important rule of substitution for variables. In the first line, he calls for the associative law and then indicates what substitutions he wants to make for A, B, and C. Here he puts 3 for A, B for B, and C + D for C, to obtain line (1) proof. Note that in the program, the call for the use of previous theorems or axioms does not itself constitute a line of proof but only the substitution instance. Further examples of this may be seen in the 15-line proof in the fifth part of fig. 2.

An important point to emphasize about the logic and algebra program is that the student is permitted to take any step in a proof that is logically valid. There is no preset enumeration of possible proofs or acceptable proofs. The program is written in recursive fashion to accept any logically valid use of the rules. The student is corrected only if he makes an error in applying the rule. However, if the student is unable to find a correct proof after a certain designated time, he is given pedagogical hints and additional help. One of our main efforts during the coming academic year will be devoted to deepening the interaction between the student and computer program when the student is having difficulty in finding a statisfactory proof. Ideally, we would like ultimately to move to a full dialogue interaction between the program and student. All of us recognize the difficulties of a completely free interaction. We do think that we can take significant steps within the technical resources now available to us.

In the fourth part of fig. 2 an example is shown of a problem-solving routine that is close in spirit to the logic and algebra program but is offered to a wider range of students. It is, I think, a universal experience in all parts of the world that students have special difficulties with "word" problems or story problems, as they are also often called. We do not yet have a very deep understanding of the reasons for the difficulties, and a primary aim of our problem-solving program is to permit a deeper investigation of the situation by isolating and, indeed, eliminating the purely computational aspects of the solution. As is illustrated in the problem about Mary's postcards, the program lists immediately after the statement of the problem each number referred to in the problem (the G at the left is for given number). The student may then request the computer to apply any one of the four rational arithmetical operations to any pair of numbers, with the exception of division by zero, or add a new line with another number, for example, to divide by 2, to change from weeks to days, etc. Thus, the student has available a simple program with 5 instructions. Line (4) of the illustration shows the form of a typical instruction: 1.2A means to add the numbers given in lines (1) and (2). The student terminates computation by inputting the line number of the answer to the problem, followed by X; here this input is 5X. This program just began running on an operational basis in January 1968, so it is as yet too early to report any results.

In fig. 3, some typical formats from the Russian course are indicated. At present this course in first-year Russian is our only effort at the university level of instruction. We may characterize the elementary teaching of Russian, or about any other foreign language, in terms of three major components. The first and most important is classroom instruction. Elementary Russian at Stanford meets five days a week for 50 minutes each day. The second major component is attendance at the language laboratory where the student hears spoken Russian and improves his own pronunciation. The third component is doing homework, which is turned in for grading and correction. In our first effort at a computer-based Russian course, we have followed the conservative course of attempting to eliminate only the first main component, namely, regular attendance in classroom. Classroom work has been totally abolished for the students taking the computer-based course. They come to the terminals, which consist of Model 35 teletypes with Cyrillic keyboards and audiotapes with earphones, for 50 min a day, five days a week, just as in the regular classrooms. The audiotapes are under computer

```
МЫ -NO ONE- _________ НЕ ПРИСТАЁМ.
TIME IS UP.  START OVER

МЫ -NO ONE- НИ К КОМУ НЕ ПРИСТАЁМ.
```

```
TYPE CORRESPONDING SENTENCES
REPLACING THE -НАДО- CONSTRUCTION WITH THE IMPERATIVE.

НЕ НАДО СМЕЯТЬСЯ
С
NO.  TRY AGAIN

НЕ СМЕЙТЕСЬ
```

```
YOU SHOULDN'T HAVE MADE EVEN ONE MISTAKE!
DON'T YOU KNOW THE BASIC FORMS -ЭТОТ, ЭТА, ЭТО-
THE FEM. ACC. -ЭТУ- AND THE ADJECTIVAL ENDINGS
 -ОГО, ОМУ, ОМ, ОЙ-?

TRY AGAIN

FILL IN.

ЭКЗАМЕН ПО ЭТ___ УРОК_.
TIME IS UP.  START OVER

ЭКЗАМЕН ПО ЭТОМУ УРОКУ.

У ТОЙ МОЛОДОЙ ДЕВУШКИ.
```

Fig. 3. Examples of exercises drawn from the computer-based Russian Course.

control and are closely keyed to all aspects of the work that is printed out on the teletypes. The exercises in fig. 3 give some typical examples of the procedure. When the student makes a mistake, as in the first exercise, he is given the opportunity to try again before he is told the correct exercise. This part of the program is similar in structure to the individualized drill-and-practice program in mathematics. When the student does not perform at a satisfactory level on standard review portions of the program, he is given additional review as exemplified in the second part of the figure. The student is first chided for not knowing the basic forms of the Russian *this* and is given additional exercises. It should be emphasized that these additional exercises are given only to those students who have not performed satisfactorily on the immediately preceding exercises. Thus they represent a typical example of the kind of contingent programming currently built into the Russian program. All phases of the program have been the primary responsibility of Professor Joseph Van Campen of the Department of Slavic Languages at Stanford. In order to get a strict comparison of the computer-based course with the regular course, both groups are being given common off-line examinations. At the end of the first term the computer-based students performed at a statistically significantly higher level. A statistic that is probably at least as important is that there was a much smaller drop-out rate from the computer-based section than from the other two sections of the course. Probably because Russian is more difficult for Americans than French, Spanish, or German, the drop-out rate has been relatively high at Stanford and other universities. We hope that we will be able to do something to change this by providing the kind of individual, concentrated attention that is given in a computer-based language course. At the university level the teaching of foreign languages will almost certainly be the first natural place for computers to be widely used, although elementary mathematics instruction at the many junior colleges in the United States will probably be a close second.

I turn now to a brief description of our most recent program, which we call dial a drill. In this program we telephone students in their homes and give them oral exercises in elementary mathematics by means of computer-generated speech. The students respond by using a touch-tone dialing pad. These devices are now standard equipment on telephones in many parts of the United States and will soon be universal. We have been particularly interested in two features of the dial-a-drill program. First, we have been concerned to find a more flexible and more easily programmed approach to audio messages than is provided by standard tape or disk, with the full message stored in analog form. For many kinds of reasons it is desirable to store, at the most, the single vocabulary-items that will be used, and then to synthesize in program the individual sentences for delivery as audio messages. Along with many other people, we are experimenting with synthesizing words, but at the present time our operational efforts do begin with the recording of words with 6-bit sampling at 6-kilocycle frequency. In the case of the dial-a-drill arithmetic program we are able to synthesize literally thousands of messages from a vocabulary of not more than 100 words. The words that are recorded are the number names *zero*, *one*, *two*, *three*. . ., *ten*, *twenty*. . ., *hundred*, the standard names for the operations, a few verbs like *equals* and a few interrogatives and related words. The intelligibility of the speech we are producing is high. We are not yet satisfied with its total quality, but it is functioning in a serviceable and practical way. At the same time, we are gaining our first operational experience in bringing computer-based curriculum into the homes of students, rather than into the schools.

Problems. In conclusion, I would like to enumerate some of the most important problems that we feel need to be solved either to permit widespread use of computer-assisted instruction, or to allow a much deeper level of interaction between student and computer program.

First of all, for widespread use we need detailed analysis and design of systems that can operate for a total cost per student that is far less than the cost of initial efforts taking place anywhere in computer-assisted instruction. In the kind of individualized drill-and-practice program I discussed, we can, with present hardware configurations and within the range of available software programs, reach for a price of $50 per student per year. In the United States the average cost of instruction is $500 per student per year. A 10 per cent charge for an individualized drill-and-practice program in elementary mathematics is too high a percentage of the total budget for mass use of computer-assisted instruction. We need to lower the cost to probably $25 or $30 a student, and ultimately would like to reach for an even lower percentage of the total operational budget allocated on a per-student basis. With the decreasing cost of computer power, with the prospect of mass production of terminals, and particularly with improved solutions of communication line problems, there is some reason for measured optimism that these costs will be brought down to the kind of figures I have mentioned within a decade.

Secondly, for deeper instructional purposes we need more powerful approaches to the store and delivery of audio messages. At present, in our Stanford laboratories, we have four separate audio systems operating. Two of them use ordinary quarter-inch tape. One uses broad-band tape containing 128 tracks with the search for a track taking place orthogonally to the direction of play. I have already mentioned our latest venture, namely, computer-generated speech. Although we hope to have a greater depth of experience and operation in this direction by next year, the present tentative outlook is that this is clearly the most promising approach but an ordinary English word is about a third to one-half second long. At 6-bit sampling and 6-kilocycle frequency, the amount of storage required for a vocabulary of 5 000 words is very high. We need either improved methods of storing and accessing or improved methods of generating words by synthesizing from below the word level. I think it is fair to say that once we have reached the promised land of computer-generated speech with all the accompanying conveniences of programming and production, it will be terribly hard to drive us back to off-line tape production and recording of audio messages. Operational and logistical headaches that are generated by tape problems are too numerous to be listed here, but I can fairly say that everyone in our operating group, ranging from linguists to programmers, is enthusiastic about pushing the computer-generated speech approach as far as we can, because of its great convenience and flexibility.

Third, I mention again the pressing problem of deepening the interaction between student and computer program in terms of open queries and constructed answers. As one of our teachers remarked not long ago, the real problem is to understand the efficient way to have the computer program ask questions, not answer them. Much of good teaching, particularly in a tutorial mode, consists of asking the right questions to guide the student's direction of thinking, not in giving categorical answers to questions generated by the student. We have just begun to work in this area ourselves and I don't think we yet have any results worth reporting.

Finally, in this short list of problems, I mention the important one of speech perception. How soon will we be able to have a genuinely voice-to-voice interaction between student and computer program? Our own efforts in this direction are definitely limited and modest. We hope in the reasonably near future to get underway a series of experiments on the problems to be faced in the real-time recognition of children responding to questions with the single spoken numerals, zero to nine. This is only a first step. We already speculate what it will be like when students will be able orally to command the computer what steps to take in constructing a mathematical proof, but we recognize fully the magnitude of the problems yet to be solved.

My list of problems is short because time and space are restricted. It would not be difficult to say more. Above all, I have not mentioned problems of system design, which revolve around the three major components of core memory size, nature of file storage and the characteristics of channel and interrupt capacities. To enter into a serious discussion of these matters would require another paper. What I have tried to do here is to give an overview of our current operations, to give some sense of what we have been able to accomplish in our initial efforts at computer-assisted instruction, and to leave you with a sense of the problems we face and must solve in taking the next steps forward.

REFERENCE

[1] P. Suppes, M. Jerman and D. Brian, Computer-assisted Instruction: Stanford's 1965-66 Arithmetic Program (Academic Press, New York, 1968) in press.

DISCUSSION

Question by H. Wurz

Is this method the best for a very bright student who may think in a different way from the computer?

Answer

This is a very serious problem. I am appalled at the amount of uniformity enforced in the classroom, and I hope the computer will do better, even if there is still a danger.

The author did not have time to reply to the following questions in detail.

Question by N. R. Saville

Since the system which is powerful enough to respond adequately can record every response of every child, is there not a considerable danger of misuse?

Question by K. Zinn

In what ways do your computer-based drill procedures differ in essential function from non-computer exercises which are administered and monitored through procedures developed for 'individually prescribed instruction' by Robert Glaser and others at the University of Pittsburgh? And how will the Stanford project employ techniques for generating or assembling material as it is needed for each student?

INVITED COMMENTARY

LESLIE D. McLEAN
The Ontario Institute for Studies in Education,
University of Toronto, Canada

Professor Suppes' paper is a notable one, for what is written there and what is not, for what is stressed and what is given cursory attention. The author is too modest, for example, in his discussion of the role of Stanford University in the development of computer-assisted instruction (CAI). Two major commercial systems, the IBM 1500 and RCA's Instructional 70, have emerged directly from research and development at Stanford and it is clear from the number and variety of students now using the various Stanford CAI systems, that here is still the major CAI effort in the world. Those of us who have had the pleasure of hearing him several times are not surprised that Professor Suppes gives more attention to the prospects than to the problems.

Notable for their absence are any references to the computer problems encountered, surpassed or bypassed. This is appropriate in a survey paper, but it surely must leave many questions in the minds of computer scientists. Perhaps someone will write a novel some day including the details of early trials, failures, successes and headaches. The integration of audio messages under computer control, for example, is still not satisfactory. The best overall interface between the computer and the audio source is still a human, though one-off black boxes abound, some of them very interesting. We wonder what devices are used in the Russian course, for example.

In terms of prospects, the major decisions are now curricular ones. What tasks are best assigned to computers? How shall we integrate these with other learning tasks? Who will prepare curricular material for presentation under computer control? How will we evaluate the quality of such material? In particular, we do not know even the major variables affecting human learning in multi-medium environments. Professor Suppes gives us only fleeting glimpses of his first curriculum solution and no look at all at his data. A frequent complaint is that Suppes and his colleagues will not stop developing often enough to analyse and report on the mountain of data they have collected. A book is promised.

Most important, what is the best role for the teacher? In the drill and practice exercises, the computer makes decisions; the teacher receives reports but has little to do but send the student back for another diagnostic test and a subsequent lesson. We have little empirical evidence to use in designing exercises at an appropriate level of difficulty. Professor Robert Stake, of the University of Illinois, has suggested that the best role for the teacher is evaluator, a constant monitor of *what* learning is taking place. The teacher may not design the curriculum but will participate in it and make fine adjustments to it in process.

In Canada, we are working on a multi-processor communications system designed to serve the student as a probe - to calculate, to test knowledge and understanding, and to retrieve information. The system has "computer", "student" and "mediator". The last is, of course, our familiar teacher in a new role. We hasten to acknowledge our debt to Professor Suppes and colleagues, though they keep moving so fast that the "standing on the shoulders of giants" metaphor is a bit too gymnastic. Inspiration and direction are present in abundance, however, and I am pleased to have had this opportunity to comment and to thank Professor Suppes for this comprehensive review of his work.